COMPUTER RESEARCH AND MODELING
2019 VOL. 11 NO. 6 P. 1017–1031
DOI: 10.20537/2076-7633-2019-11-6-1017-1031

КиМ

THE 3RD BRICS MATHEMATICS CONFERENCE

UDC: 519.6

Quadratic Padé Approximation: Numerical Aspects and Applications

M. Fasondini[1,a], N. Hale[2,b], R. Spoerer[2,c], J. A. C. Weideman[2,d]

[1]School of Mathematics, Statistics and Actuarial Science,
Sibson Building, Parkwood Road, University of Kent,
Canterbury, Kent, CT2 7FS, United Kingdom
[2]Department of Mathematical Sciences, Stellenbosch University,
Stellenbosch 7600, South Africa

E-mail: [a] mfasondini@gmail.com, [b] nickhale@sun.ac.za, [c] rene@rspoerer.com, [d] weideman@sun.ac.za

Received 31.05.2019.
Accepted for publication 14.11.2019.

Padé approximation is a useful tool for extracting singularity information from a power series. A linear Padé approximant is a rational function and can provide estimates of pole and zero locations in the complex plane. A quadratic Padé approximant has square root singularities and can, therefore, provide additional information such as estimates of branch point locations. In this paper, we discuss numerical aspects of computing quadratic Padé approximants as well as some applications. Two algorithms for computing the coefficients in the approximant are discussed: a direct method involving the solution of a linear system (well-known in the mathematics community) and a recursive method (well-known in the physics community). We compare the accuracy of these two methods when implemented in floating-point arithmetic and discuss their pros and cons. In addition, we extend Luke's perturbation analysis of linear Padé approximation to the quadratic case and identify the problem of spurious branch points in the quadratic approximant, which can cause a significant loss of accuracy. A possible remedy for this problem is suggested by noting that these troublesome points can be identified by the recursive method mentioned above. Another complication with the quadratic approximant arises in choosing the appropriate branch. One possibility, which is to base this choice on the linear approximant, is discussed in connection with an example due to Stahl. It is also known that the quadratic method is capable of providing reasonable approximations on secondary sheets of the Riemann surface, a fact we illustrate here by means of an example. Two concluding applications show the superiority of the quadratic approximant over its linear counterpart: one involving a special function (the Lambert W-function) and the other a nonlinear PDE (the continuation of a solution of the inviscid Burgers equation into the complex plane).

Keywords: Padé approximation, numerical singularity detection

Citation: *Computer Research and Modeling*, 2019, vol. 11, no. 6, pp. 1017–1031.

MF acknowledges financial support from the EPSRC grant EP/P026532/1. The research of NH was supported by the National Research Foundation (NRF) of South Africa (Grant Number 109210). RS was supported by the NRF via the research grants of NH and JACW. JACW acknowledges additional support from the H. B. Thom Foundation at Stellenbosch University. Nick Trefethen took a keen interest in this work and made many useful suggestions.

1 Introduction

In many applications of complex variables it is necessary to extract singularity information of a function, given only a power series representation of that function. Any computation based on a direct summation of the power series is bound to fail outside the radius of convergence of the series. To extend the domain of approximation, and in particular to extract singularity information, different techniques are required. Padé approximation is one of the prominent methods of achieving this. Standard (linear) Padé approximation leads to a meromorphic approximant, from which pole information can be extracted. Quadratic Padé approximation, by contrast, leads to a two-sheeted approximant from which additional information can be extracted. First introduced to the numerical analysis literature in [Shafer, 1974], this quadratic approximation, and in particular its numerical computation, is the focus of the present study.

Consider a function f assumed to be analytic in an open neighbourhood of the origin, admitting the formal series expansion

$$f(z) = \sum_{k=0}^{\infty} f_k z^k. \tag{1}$$

The linear (n, n) Padé approximation to f is defined by the polynomials[1]

$$p(z) = \sum_{k=0}^{n} p_k z^k, \qquad q(z) = \sum_{k=0}^{n} q_k z^k, \tag{2}$$

that satisfy

$$p(z) + q(z)f(z) = \mathcal{O}(z^{2n+1}), \qquad z \to 0. \tag{3}$$

This idea can be extended to the quadratic (n, n, n) case, which is defined by polynomials

$$p(z) = \sum_{k=0}^{n} p_k z^k, \qquad q(z) = \sum_{k=0}^{n} q_k z^k, \qquad r(z) = \sum_{k=0}^{n} r_k z^k, \tag{4}$$

such that

$$p(z) + q(z)f(z) + r(z)f(z)^2 = \mathcal{O}(z^{3n+2}), \qquad z \to 0. \tag{5}$$

In both (3) and (5) there is nonuniqueness in that the polynomials are defined only up to a multiplicative constant. This can be eliminated by fixing one of the coefficients or enforcing some other constraint.

Dropping the order terms on the right, equations (3) and (5) define, respectively, the (n, n) linear approximant F_1 and the (n, n, n) quadratic approximants F_+ and F_-:

$$F_1(z) = -\frac{p(z)}{q(z)}, \qquad F_\pm(z) = \frac{-q(z) \pm \sqrt{d(z)}}{2r(z)}. \tag{6}$$

The discriminant polynomial, d, in the latter formula is given by

$$d(z) = q(z)^2 - 4p(z)r(z). \tag{7}$$

These approximants can be used to approximate the locations of zeros and singularities of the function f. The linear approximant F_1 has only zeros and poles, at most n of each. The quadratic approximants $F_\pm$ have zeros, poles, and square root branch points. From the identity $F_+F_- = p/r$ one concludes that the zeros of the approximant are defined by the roots of p, provided these roots are distinct from the roots of r. There are therefore at most n zeros, split between F_+ and F_-. The same

[1] Approximants (m, n) with $m \neq n$ are defined similarly but will not be considered here.

arguments apply to the poles of the approximant, with the roles of p and r reversed. Finally, square root branch points of the approximant are defined by the roots of d, provided they are of odd multiplicity. There are at most $2n$ of these branch points, some of which may approximate the actual branch points of f and others that may be spurious.

Cubic and higher degree approximations can be defined analogously to (3) and (5), which might have advantages in the approximation of algebraic branch points of higher order. For examples of approximating functions with logarithmic branch points, see [Driscoll, Fornberg, 2001]. These are examples of the more general Hermite – Padé approximation process [Baker, Graves-Morris, 1996], but only the quadratic case (5) will be considered here. Padé and Hermite – Padé approximation can also be extended from the power series (1) to Chebyshev series (on a bounded interval; see [Boyd, 2009]), or to Fourier series (on a periodic interval; see Section 6.2). Alternatively, Hermite – Padé approximants can be defined in terms of function values rather than expansion coefficients by generalizing the rational least squares approach of [Gonnet et al., 2011].

The objective here is to see whether the quadratic approximant succeeds in capturing singularity information of the original function in a domain of reasonable size about the origin, using only polynomials of relatively low degree. Other computational studies might consider convergence rates at fixed values of z as n increases, or computing zero, pole and branch point locations to many digits of accuracy, but that is not our primary focus.

Complementary to the objective mentioned in the previous paragraph, we also address numerical aspects of quadratic approximations in this paper. Our interest in a floating-point implementation rather than using symbolic software stems from those applications where the singularity distribution is parameter dependent. If such parameters range over many thousands of values, the computational speed offered by floating-point arithmetic is essential.

It is well known that linear Padé approximations can have spurious poles, i.e., poles that do not correspond to any singularities of f [Stahl, 1998]. Here we show that, analogously, the quadratic approximant can have spurious branch points, which causes the quadratic equation (5) to be ill-conditioned. Likewise, if linear systems are used to solve the linear or quadratic problems, the matrices involved can be highly ill-conditioned. In the linear case it is known that despite these large condition numbers, the Padé approximation itself can be evaluated to high relative accuracy [Luke, 1980]. We extend these ideas here to the quadratic case.

The outline of the paper is as follows: In Section 2 we present an example of the approximation of a simple cube root function. It highlights some of the strengths of the quadratic approximant, for example, high accuracy on the principal sheet and even moderate accuracy on the secondary sheets. It also highlights some of its weaknesses, for example, possible ill-conditioning of the quadratic equation that defines the approximant. In Sections 3–5 we turn to numerical aspects: The standard method of computing the polynomials in the quadratic approximant by solving linear systems is discussed, as well as the conditioning of these systems. An alternative recursive method is also discussed and some of its advantages pointed out. Section 6 is devoted to two applications.

2 Approximation on a Riemann Surface: An Example

To demonstrate the approximation capabilities of the quadratic approximant, we consider the simple cube root function

$$f(z) = (1+z)^{1/3} = \sum_{k=0}^{\infty} \binom{\frac{1}{3}}{k} z^k, \tag{8}$$

where the series converges in the open unit disc centred at $z = 0$. This is a function on a three-sheeted Riemann surface, displayed in Figure 1, which we are going to approximate with a two-sheeted surface,

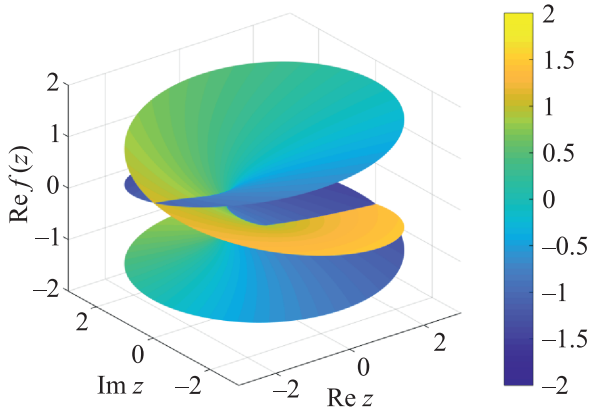

Figure 1. Representation of the three-sheeted Riemann surface of $f(z) = (1+z)^{1/3}$. Here the vertical axis represents the real part and colour the imaginary part as described in [Corless, Jeffrey, 1998]. The two-sheeted quadratic Padé approximation to this surface is shown below in Figure 5

as defined by the quadratic approximant. We shall take the domain of interest D somewhat arbitrarily as the disc of radius 3 centred at the origin (Figures 1 and 5) or the square $[-3,3]\times[-3,3]$ (Figures 3 and 4).

In our first comparison we restrict the domain to be the real interval $[-3,3]$ and compare the accuracy of the linear and quadratic Padé approximations to the principal branch of the function (the top sheet in Figure 1). Figure 2 shows relative errors, where the first 17 coefficients in the series in (8) were used to construct the $(8,8)$ and the $(5,5,5)$ approximants. To choose between F_+ and F_-, we have used the value closest to the principal branch value of $(1+z)^{1/3}$. When the function is known only through its power series, this will not be possible and the choice will have to be made on other grounds; see Section 5.1.

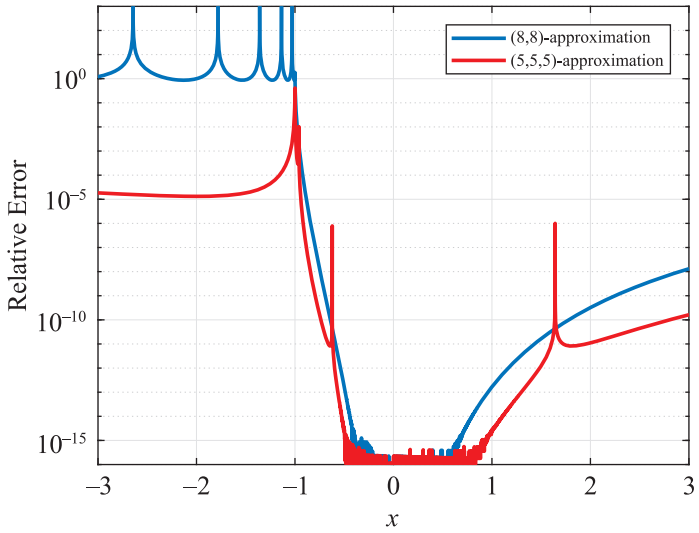

Figure 2. Errors in the linear and quadratic Padé approximations of the function $f(z) = (1+z)^{1/3}$ (principal branch). The superiority of the quadratic approximation is most prominent on the branch cut $(-\infty,-1]$. (Note that the spikes in the blue curve reach all the way to infinity because these are poles of the linear approximant. In the red curve they have finite maxima because they are caused by spurious branch points in the quadratic approximant. In fact, at each of these points there is a pair of nearby roots of d; see Section 5.2.)

Near the origin in Figure 2, both linear and quadratic approximations achieve machine accuracy (relative error $\sim 10^{-16}$ in IEEE arithmetic), as represented by the highly oscillatory error curve. As one goes out on the positive real axis, the quadratic approximant is about two orders of magnitude better than the linear one. But the real advantage is on the branch cut $(-\infty, -1]$ where the quadratic approximant is about five orders of magnitude better. (When n is increased from 5 to 7, another one to two orders of magnitude are gained.) It should be noted that the linear approximant achieves no accuracy at all on $(-\infty, -1]$, because its poles are located on the branch cut as represented by the spikes in the error curve. For the theory behind this phenomenon we refer to [Stahl, 1997]. There are also spikes in the error curve of the quadratic approximant, but these are not caused by poles. These correspond to spurious branch points, i.e., branch points of the approximant that are not in any sense an approximation to the physical branch points of the original function. The instability associated with these points is discussed further in Sections 4 and 5.2.

Next, we repeat the above calculation but compare accuracy of the linear and quadratic approximants in the complex domain D; see Figure 3. Again the superiority of the quadratic approximant is most striking near the branch cut.

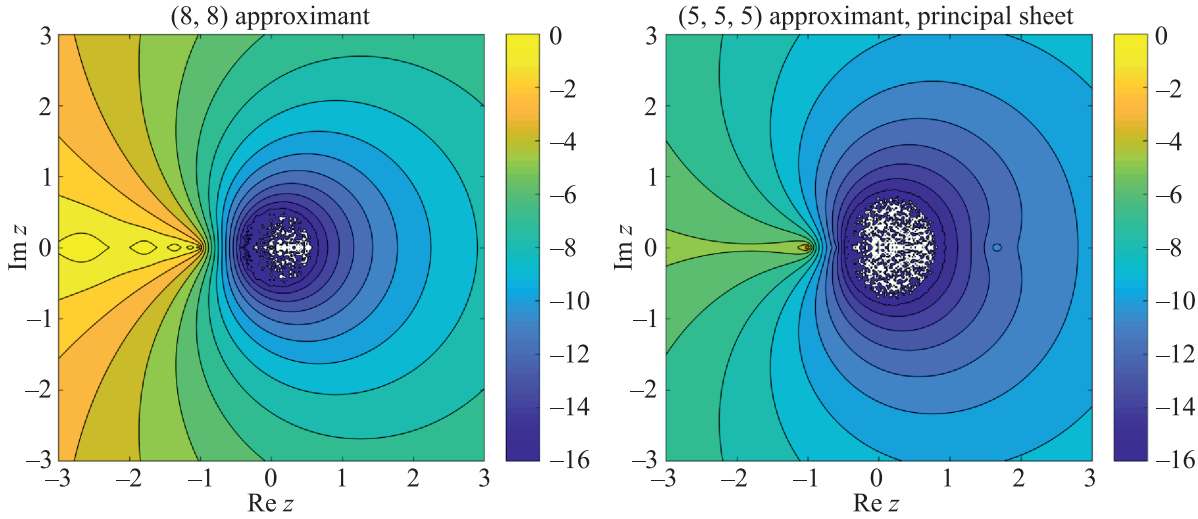

Figure 3. Same as Figure 2, but in the complex plane. The $\log_{10}$ of the relative error in the principal branch is shown as a contour plot

What about the accuracy of the other branch of the quadratic approximant? The linear approximant is single-valued and can only approximate the principal branch. As noted in [Short, 1979], however, quadratic approximants are capable of approximating secondary branches. Two- and three-dimensional displays of the error are shown, respectively, in Figures 4 and 5. It is noted that the accuracy of the approximation on the secondary branch is poorer than on the principal branch. This is due to the fact that the second sheet is effectively farther from the point of expansion.

The function f has a single branch point (and a zero) at $z = -1$ and is pole-free, whereas, as noted above, the quadratic approximant can have as many as $2n$ branch points, n poles and n zeros, and this is indeed the case for the $(5, 5, 5)$ approximant of (8). On the displayed domains, the branch points are at[2]

$$-0.99914769,\ -0.96077940 \pm 0.00020391i,\ -0.62144269 \pm 0.00000014,\ 1.64160798 \pm 0.00000010. \tag{9}$$

[2] The close proximity of some of these roots is not a consequence of floating-point error. The roots were checked to high precision using symbolic software.

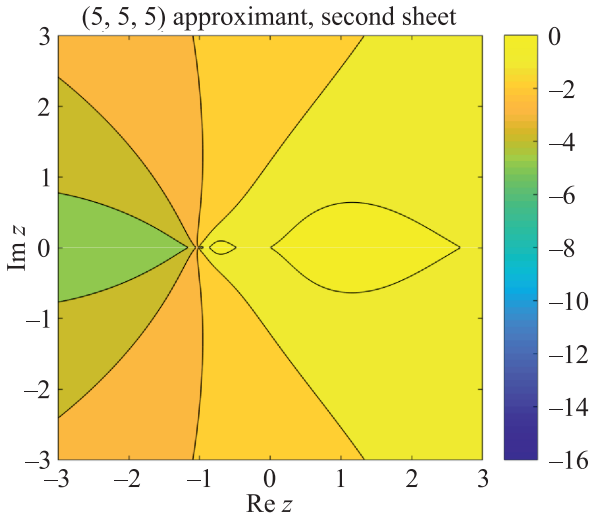

Figure 4. Same as Figure 3, but the relative error on the secondary branch is shown. Only about five-digit accuracy is achieved, near the negative real axis. By contrast, the accuracy in the principal branch ranges between five and sixteen digits as shown in Figure 3.

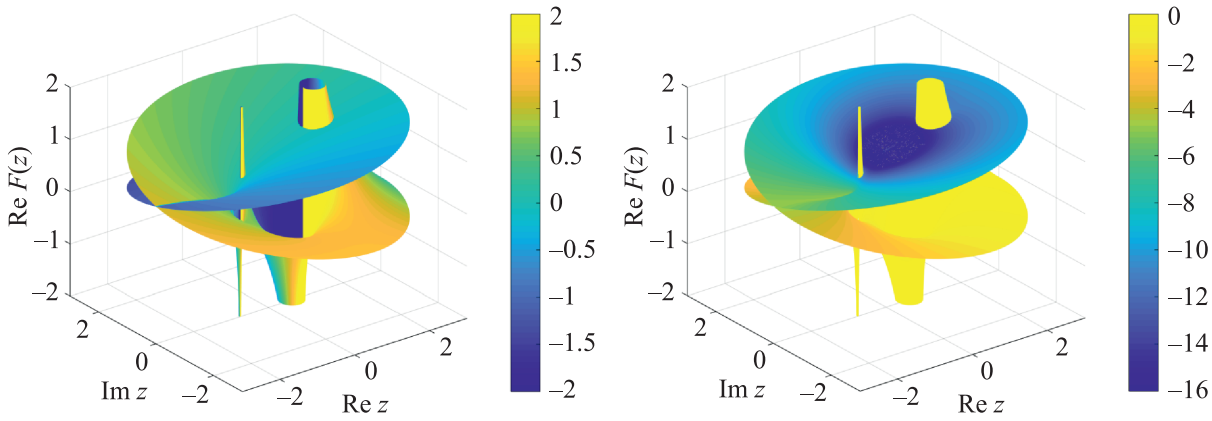

Figure 5. Representation of the two-sheeted Riemann surface of the $(5, 5, 5)$ appproximation to the function $f(z) = (1 + z)^{1/3}$. Left: colour is used to represent the imaginary part, same as in Figure 1. Right: colour is used to represent the $\log_{10}$ of the relative error. The two protruding peaks correspond to poles on the secondary sheet; see the discussion below (10).

These points correspond to the upward spikes in Figure 2, where the approximant loses accuracy. This will be discussed further in Sections 4 and 5.2. The poles and zeros of the approximant are, respectively,

$$-0.97573796,\ -0.72940734,\ 0.90299818; \tag{10}$$

and

$$-0.99848792,\ -0.93924683,\ -0.47451342,\ 2.69559169.$$

All of these poles/zeros are simple and all are located on the second sheet. The remaining three branch points, two poles and one zero are outside the domains shown. Note that only the second and third poles in (10) are visible in Figure 5. This is because the first pole has a small residue, 0.0059, and consequently it is so localised that it is not visible on the discrete grid of function values used to plot the Riemann surface. The residues of the second and third poles in (10) are larger by comparison, namely, 0.099 and 1.3.

We now turn to the numerical computation of Padé approximants such as those used in this section.

3 Computing the coefficients

3.1 Solving linear systems

The method of computing the coefficients of the linear Padé approximant by solving a linear system is described in [Baker, Graves-Morris, 1996, Sect. 1.1]. Following a similar approach for the quadratic approximant, we substitute the power series (1) and (4) into the quadratic Padé equation (5), to find

$$\sum_{j=0}^{n} p_j z^j + \sum_{j=0}^{n} q_j z^j \sum_{k=0}^{\infty} f_k z^k + \sum_{j=0}^{n} r_j z^j \left(\sum_{k=0}^{\infty} f_k z^k \right)^2 = \mathcal{O}(z^{3n+2}). \tag{11}$$

Using the well-known convolution property of polynomial multiplication, this becomes

$$\sum_{j=0}^{n} p_j z^j + \sum_{k=0}^{\infty} \Big(\sum_{j=0}^{\min(k,n)} f_{k-j} q_j \Big) z^k + \sum_{k=0}^{\infty} \Big(\sum_{j=0}^{\min(k,n)} f_{k-j}^2 r_j \Big) z^k = \mathcal{O}(z^{3n+2}), \tag{12}$$

where f_ℓ^2 is the ℓ-th coefficient in the formal power series of $f^2(z)$, which is itself given by the discrete convolution

$$f_\ell^2 = \sum_{j=0}^{\ell} f_{\ell-j} f_j. \tag{13}$$

Equating the first $3n+2$ terms on the left and right of (12) and writing $\boldsymbol{p} = [p_0, p_1, \ldots, p_n]^T$, etc., one obtains the linear system of equations

$$\begin{bmatrix} I & \begin{bmatrix} C_n \end{bmatrix} & \begin{bmatrix} C_n^2 \end{bmatrix} \\ 0 & & \\ 0 & & \end{bmatrix} \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \\ \boldsymbol{r} \end{bmatrix} = \mathbf{0}. \tag{14}$$

I is the $(n+1)\times(n+1)$ identity matrix, C_n is the first $3n+2$ rows and $n+1$ columns of the Toeplitz matrix

$$C = \begin{bmatrix} f_0 & & & \\ f_1 & f_0 & & \\ f_2 & f_1 & f_0 & \\ \vdots & \vdots & \ddots & \ddots \end{bmatrix}, \tag{15}$$

and C_n^2 is the first $3n+2$ rows and $n+1$ columns of C^2.

Notice that the matrix in (14) is of dimension $(3n+2)\times(3n+3)$ and so the linear system is under-determined, which relates to the arbitrary scaling of p, q, and r in (5). Choosing the conventional normalisation $r_0 = 1$, one may take the appropriate column of C_n^2 to the right and solve the resulting square system using standard techniques (e.g., Gaussian elimination). Similarly to the linear case described in [Baker, Graves-Morris, 1996, Sect. 1.1], efficiency and stability can be improved by first solving for $\boldsymbol{q}$ and $\boldsymbol{r}$ using the $(2n+1)\times(2n+2)$ lower-right subblock of the matrix in (14) and then recovering $\boldsymbol{p}$ from the result.

As an alternative, one can instead compute the required null vector of the over-determined linear system (14) (or its lower-right subblock) by using the singular-value decomposition (SVD). One advantage of this approach is that it will not break down in instances where the normalisation $r_0 = 1$

fails (i.e., when r vanishes at the origin)[3]. In the case of linear Padé approximation, a second advantage of the SVD approach is that it may be used to address the occurrence of "Froissart doublets" (nearby pole/root pairs which should cancel mathematically, but in floating-point arithmetic do not). This idea was introduced in [Gonnet et al., 2013] for the linear Padé approximant, but its generalization to quadratic and higher order Padé approximants will not be considered here.

3.2 Recursive algorithms

The linear system approach of the previous section is not the only algorithm available for computing linear and quadratic approximants. Alternative algorithms, also for computing more general Hermite – Padé approximants, are described in [Derksen, 1994; Loi, McInnes, 1984; Sergeyev, 1986] and elsewhere. We briefly describe Sergeyev's algorithm, simplified to the quadratic case.

Given a function f defined by the power series (1), the algorithm recursively computes quadratic Padé approximations of the form

$$S_k(z) \coloneqq P_k(z) + Q_k(z)f(z) + R_k(z)(f(z))^2 = \mathcal{O}(z^k) \tag{16}$$

for $k = 0, 1, \ldots$ by using the residuals $S_k(z) = z^k(\alpha_k + \beta_k z + \gamma_k z^2 + \ldots)$. It is easy to verify that (16) is satisfied for $k = 0, 1, 2$ by setting

$$\begin{aligned}
&P_0(z) = 1, && Q_0(z) = 0, && R_0(z) = 0, && S_0(z) = 1;\\
&P_1(z) = -f_0, && Q_1(z) = 1, && R_1(z) = 0, && S_1(z) = f(z) - f_0;\\
&P_2(z) = f_0^2, && Q_2(z) = -2f_0, && R_2(z) = 1, && S_2(z) = \big(f(z) - f_0\big)^2.
\end{aligned}$$

For $k > 2$, the polynomials $P_k(z), Q_k(z), R_k(z)$ and the residuals $S_k(z)$ can all be computed from the above starting values and the following four-term recursion:

$$s_k(z) = (\alpha_{k-3}\beta_{k-2} - \alpha_{k-2}\beta_{k-3})s_{k-1}(z) - \alpha_{k-3}\alpha_{k-1}s_{k-2}(z) + \alpha_{k-2}\alpha_{k-1}zs_{k-3}(z).$$

Note that P_0,P_1,P_2 are constants; P_3,P_4,P_5 are linear polynomials; P_6,P_7,P_8 are quadratic polynomials; and so forth. A similar pattern holds for the Q_k and the R_k. Here the first linear polynomials are Q_4 and R_5. The polynomials p, q, and r defined by (5) can thus be obtained by calculating P_{3n+2}, Q_{3n+2}, and R_{3n+2}. In particular, it should be noted that the above recursion ensures that $S_k(z) = \mathcal{O}(z^k)$.

One advantage of the recursive algorithms is the fact that they are often touted to be of lower computational complexity than the linear system approach. This advantage is seldom realised in practical floating-point calculations, however, because the degrees of the polynomials in practice hardly ever exceed twenty or thirty (for larger degrees, numerical stability is a bigger concern than computational complexity; see Section 4). There is, however, a very real advantage of the recursive algorithms, namely, in the detection of spurious branch points. Because the polynomials P_k, Q_k, R_k are built up with increasing degree, branch points can be monitored by computing the roots of the corresponding discriminant polynomials. Those that converge to a fixed limit are possible candidates for physical branch points and the others are likely to be spurious.

We conclude this section by comparing the accuracy of the linear system approach with that of the Sergeyev algorithm outlined above. To compare the two approaches, we first compute the exact Padé polynomials by solving the linear system of Section 3.1 using symbolic software. Then the polynomials are recomputed using the two methods in IEEE arithmetic.

As will be discussed in Section 4, it is not useful to compare these polynomials directly. Rather, at certain chosen points in the complex plane, we will compare the symbolically computed

[3] One example is $f(z) = \log(1 + z)$, for odd values of n.

approximant $F_\pm$ with the numerical counterparts $F_{\pm,L}$ and $F_{\pm,S}$, which are computed using the linear system and the Sergeyev algorithm, respectively. In each case, for consistency, we will choose the branch of the approximant which is closest to the function f.

Table 1 shows such a comparison for (n,n,n) Padé approximants of the function $f(z) = (1+z)^{1/3}$. The approximants computed using the linear system are somewhat more accurate than those computed using the Sergeyev algorithm, but the difference is never great. This trend seems to hold with other functions f and other values of z.

Table 1. A comparison between the accuracy of quadratic approximants as computed by (S) the Sergeyev algorithm and (L) the linear system approach. The function is $f(z) = (1+z)^{1/3}$.

	$z=1$		$z=2$		$z=i$	
n	$\lvert F_{\pm,L} - F_\pm\rvert$	$\lvert F_{\pm,S} - F_\pm\rvert$	$\lvert F_{\pm,L} - F_\pm\rvert$	$\lvert F_{\pm,S} - F_\pm\rvert$	$\lvert F_{\pm,L} - F_\pm\rvert$	$\lvert F_{\pm,S} - F_\pm\rvert$
4	0.0	6.7×10^{-16}	2.2×10^{-16}	6.7×10^{-16}	5.6×10^{-17}	6.7×10^{-16}
8	2.2×10^{-16}	2.2×10^{-16}	4.4×10^{-16}	3.8×10^{-15}	2.5×10^{-16}	6.9×10^{-16}
12	0.0	1.8×10^{-14}	4.2×10^{-15}	2.6×10^{-14}	2.3×10^{-16}	4.9×10^{-14}
16	4.4×10^{-16}	1.1×10^{-13}	5.3×10^{-15}	2.2×10^{-15}	2.2×10^{-16}	9.5×10^{-14}
20	6.7×10^{-16}	1.1×10^{-14}	2.2×10^{-15}	3.4×10^{-14}	5.0×10^{-16}	7.3×10^{-14}

4 Conditioning of the linear systems

The linear systems mentioned in Section 3.1 are typically ill-conditioned, as demonstrated in Table 2. Because of this ill-conditioning, large relative errors can be expected in the computed coefficients. One finds, however, that the Padé approximant can nevertheless be evaluated to high relative accuracy (as already observed in Table 1).

Table 2. 2-norm condition numbers of the linear systems for solving the (n,n) linear and (n,n,n) quadratic Padé approximation problems for the function $f(z) = (1+z)^{1/3}$, where n is such that the total number of degrees of freedom in each case is ℓ. (For example, the $\ell = 5$ column corresponds to $(2,2)$ and $(1,1,1)$ approximants, and the $\ell = 11$ column to $(5,5)$ and $(3,3,3)$.) We present the condition number of the square lower-right subblock of the matrix in (14) after normalising such that $r_0 = 1$, as described in 3.1

ℓ	5	11	17	23
Linear	1.7×10^{1}	3.5×10^{5}	1.1×10^{10}	3.5×10^{14}
Quadratic	3.0×10^{2}	4.4×10^{7}	7.3×10^{12}	1.5×10^{17}

The explanation of [Luke, 1980] for the linear case is based on a perturbation analysis: Consider $F_1 = -p/q$, and perturb it to $\widetilde{F_1} = -(p+\delta p)/(q+\delta q)$. Taylor expansion, to first order in δp and δq, leads to

$$\widetilde{F_1}(z) \approx F_1(z) + \frac{\partial F_1}{\partial p}\delta p + \frac{\partial F_1}{\partial q}\delta q. \tag{17}$$

The partial derivatives are computed by setting the right-hand side of (3) to zero, and differentiating the left-hand side implicitly

$$\frac{\partial F_1}{\partial p} = \frac{-1}{q}, \qquad \frac{\partial F_1}{\partial q} = \frac{-F_1}{q}. \tag{18}$$

By substituting (18) into (17) one obtains

$$\frac{F_1(z) - \widetilde{F_1}(z)}{F_1(z)} \approx \frac{\varrho_1(z)}{q(z)F_1(z)}, \qquad \varrho_1(z) \coloneqq \delta p(z) + \delta q(z)F_1(z). \tag{19}$$

One concludes that if ϱ_1 is small, then the relative error is small, except near the zeros of F_1. Luke argues that ϱ_1 is small even if δp and δq are relatively large, by showing that $\varrho_1 = \mathcal{O}(z^{n+1})$, $z \to 0$. Numerical computations are given in [Luke, 1980] to support the theory.

In the quadratic case, the partial derivatives analogous to (18) follow from implicit differentiation of (5):

$$\frac{\partial F_\pm}{\partial p} = \frac{-1}{q + 2rF_\pm}, \quad \frac{\partial F_\pm}{\partial q} = \frac{-F_\pm}{q + 2rF_\pm}, \quad \frac{\partial F_\pm}{\partial r} = \frac{-F_\pm^2}{q + 2rF_\pm}. \tag{20}$$

Using a Taylor approximation analogous to (17) and the fact that $q + 2rF_\pm = \pm\sqrt{d}$, one obtains

$$\frac{F_\pm(z) - \widetilde{F}_\pm(z)}{F_\pm(z)} \approx \frac{\pm\varrho_\pm(z)}{\sqrt{d(z)}\,F_\pm(z)}, \qquad \varrho_\pm(z) \coloneqq \delta p(z) + \delta q(z)F_\pm(z) + \delta r(z)F_\pm(z)^2. \tag{21}$$

By similar reasoning as in [Luke, 1980], it follows that $\varrho_\pm(z) = \mathcal{O}\left(z^{n+1}\right)$, $z \to 0$. When $\varrho_\pm$ is small the relative error is small, except near the zeros of $F_\pm$ and near the zeros of the discriminant polynomial d. The latter zeros correspond to branch points of $F_\pm$. These points have already been observed to be troublesome in Figure 2.

The validity of the estimate (21) is confirmed by the contour plots in Figure 6, which shows the two quantities either side of the $\approx$ sign in the complex plane. We could compute the quantities δp, δq and δr for this simple function by computing p, q and r to high precision in symbolic software. The effect of the factor $\sqrt{d(z)}$ in (21) can be seen at certain locations of the plot, for example, near $z = 1.64$ and $z = -0.96$; see (9).

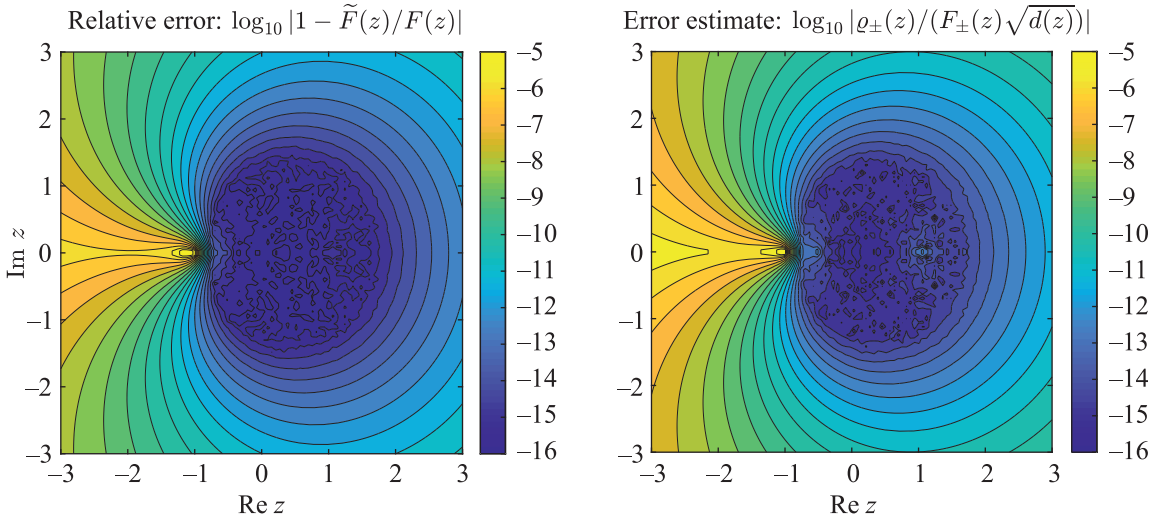

Figure 6. Left: Contour plot of the relative difference between $F_\pm(z)$ and $\widetilde{F}_\pm(z)$, where the former is the $(5, 5, 5)$ quadratic approximant as computed with exact polynomials p, q and r (available from [Sergeyev, 1986]) and the latter is the corresponding quadratic approximant computed by solving the linear system (14). The function is $f(z) = (1 + z)^{1/3}$, and this formula was used in choosing between $F_+(z)$ and $F_-(z)$ at each value of z. Right: Estimated relative difference based on (21). Although the relative errors in the computed p, q, and r are large in this region ($\sim 10^{-5}$, not shown), $F_\pm(z)$ itself has good relative accuracy, as predicted by (21)

5 Evaluating the quadratic approximation

Having computed the polynomials that define the quadratic approximant, the quadratic formula in (6) has to be evaluated. This involves two steps: (a) choosing between F_+ and F_-, and (b) implementing the formula so that it is stable against floating-point cancellation error.

5.1 *Branch cuts*

The quadratic formula in (6) is double-valued. If a single-valued approximant in the desired domain is required, branch cuts have to be introduced. This choice is up to the user and involves among other things the computation of the roots of the discriminant (7) and then distinguishing between physical and spurious branch points. For large n this is no simple task, but this may be simplified by the observation that the spurious ones often occur in pairs; compare for example (9). In addition, if a recursive algorithm such as the one of Section 3.2 is used, it is possible to detect spurious branch points as described in that section.

Here we discuss a particular choice of branch cuts, which may not be practical in all situations but it has a well-defined meaning as described below. Namely, we let the linear approximant dictate where the branch cuts should go, by picking the value of F_+ or F_- closest to the value of F_1, i.e.,

$$F_2(z) = \begin{cases} F_+(z) & \text{if } |F_+(z) - F_1(z)| \le |F_-(z) - F_1(z)|, \\ F_-(z) & \text{otherwise.} \end{cases} \tag{22}$$

It is well known that if a function is multivalued, then the linear approximant will model this by putting poles and zeros on special curves emanating from the branch points. These curves are sets of minimal capacity as defined in [Stahl, 1997]. It is expected that the choice (22) will put the branch cuts in the quadratic approximant close to these sets of minimal capacity. To demonstrate this, we take an example from [Stahl, 1997], namely,

$$f(z) = \sqrt{1 - \frac{2}{z^2} + \frac{9}{z^4}}, \tag{23}$$

which has four branch points, at $z = \pm\sqrt{2} \pm i$. We let $\zeta = 1/z^2$, and compute the first 29 coefficients of $(1 - 2\zeta + 9\zeta^2)^{1/2}$ to create the $(14, 14)$ linear approximation and the $(9, 9, 9)$ quadratic approximation. (Using lower degree polynomials does not resolve the branching structure adequately.) The results, shown as phase plots[4], can be seen in Figure 7. This should be compared to Figure 1 in [Stahl, 1997].

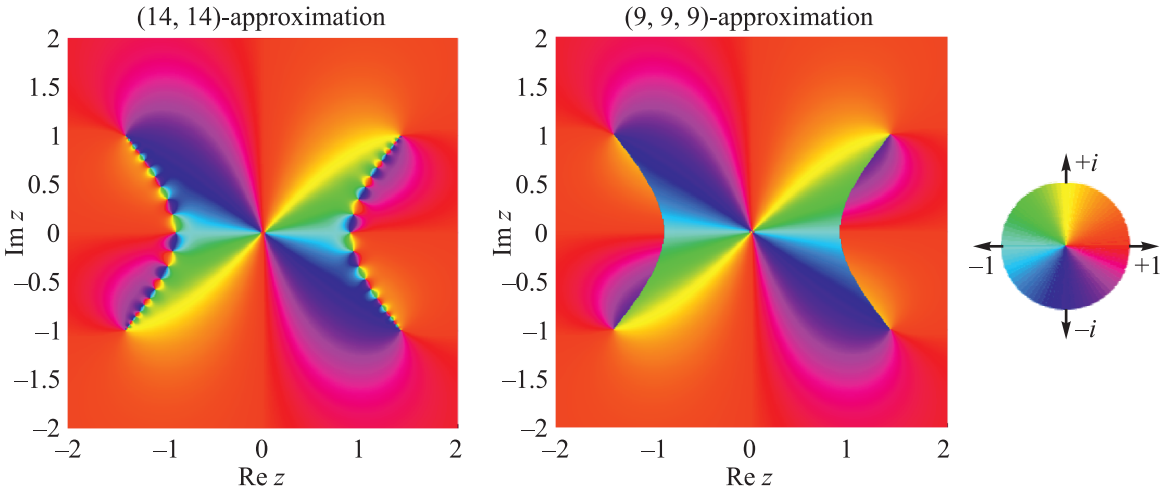

Figure 7. Left: Phase plot of the linear approximant to (23). The poles (and zeros) line up on curves of mimimal capacity as defined in [Stahl, 1997]. Middle: Quadratic approximant, with branch cuts determined from the linear approximant. Right: The colour wheel according to which the phase $\phi(z) \in [-\pi, \pi)$ of a function $f(z) = |f(z)| \exp(i\phi(z))$ is indicated in a phase plot. (The figure is taken from `https://dlmf.nist.gov/help/vrml/aboutcolor`.)

[4] Phase plots are described here: `http://www.visual.wegert.com/`.

5.2 Evaluating the quadratic formula

The quadratic formula in (6) can be ill-conditioned when the discriminant polynomial (7) vanishes, i.e., near branch points of $F_\pm$, see (21). It can also suffer from floating-point cancellation, whenever $4|pr| \ll |q|^2$, which happens near roots/poles of $F_\pm$. While the problem of ill-conditioning cannot be avoided, the numerical instability can be avoided by the standard techniques described in undergraduate numerical analysis courses. That is, one first evaluates F_+ and F_- as in (6) and then uses the identity $F_+F_- = p/r$ to set

$$F_\pm(z) = \begin{cases} F_\pm(z), & \text{if } |F_\pm(z)| \geq |F_\mp(z)|, \\ p(z)/(r(z)F_\mp(z)), & \text{otherwise.} \end{cases}$$

This formula is based on the fact that floating-point cancellation in the formula (6) more acutely affects the smaller of the two values of $F_+(z)$ and $F_-(z)$.

The destructive effect of spurious branch points can be seen in Figures 2, 3, and 6. First, in Section 4 we saw that large relative errors can be expected where the discriminant polynomial vanishes. Second, at these points the quadratic equation defined by (5) has a root of double multiplicity, and rootfinding is known to be ill-conditioned in this situation. The combination of these two effects means that in the vicinity of the branch points (spurious or otherwise) degradation in accuracy should be expected. This loss of accuracy can be seen as the upward spikes in Figure 2, as well as less clearly in the right frames of Figures 3 and 6.

6 Applications

6.1 The Lambert W-function

The Lambert function $w = W(z)$ is defined by the equation $we^w = z$, and it has the power series

$$w = z\sum_{k=0}^{\infty} \frac{(-1-k)^k}{(k+1)!} z^k, \tag{24}$$

which converges for $|z| < 1/e$ [Corless et al., 1996]. We approximate the principal branch of this function, which has a square root branch point at $z = -1/e$. Computing linear and quadratic approximants from the first 17 terms of (24) yields the errors shown in Figure 8. The quadratic approximant improves on the linear approximant by several orders of magnitude, particularly near the branch cut $(-\infty, -1/e]$.

6.2 The Burgers equation

In [Bessis, Fournier, 1990] the authors investigated the Riemann surface associated with a shock in the inviscid Burgers equation

$$u_t + uu_x = 0, \quad u(x,0) = u_0(x), \quad -\infty < x < \infty. \tag{25}$$

By choosing the initial condition, u_0, to be a cubic polynomial, they could apply Cardano's formula to the implicit solution formula $u = u_0(x - ut)$. Thus they showed that, for $t > 0$, the solution has a conjugate pair of branch point singularities on the imaginary axis. As t increases these branch points travel along the imaginary axis until they meet on the real axis at the instant of shock formation.

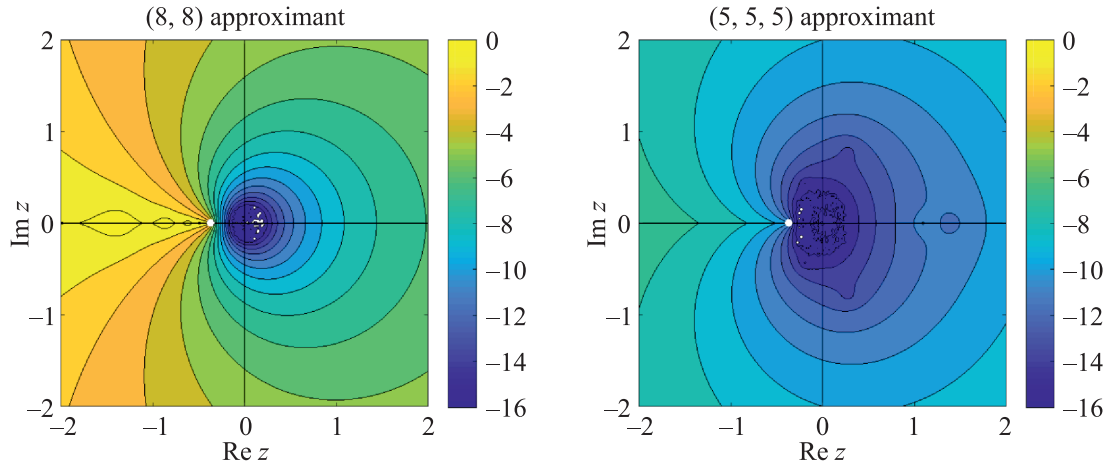

Figure 8. Left: Contour plot of $\log_{10}$ of the relative error in the $(8,8)$ linear Padé approximant for the Lambert W function (principal branch). Right: Same for the $(5,5,5)$ quadratic Padé approximant. Each case corresponds to 17 degrees of freedom, i.e., the number of nonzero terms used in the power series expansion. The white dot represents the branch point at $z=-1/e$

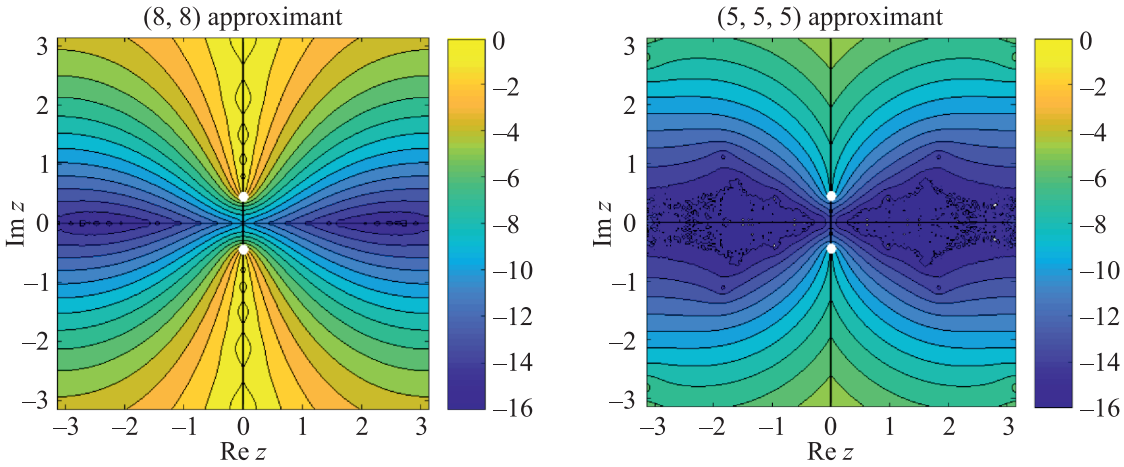

Figure 9. Same as Figure 8, but the function is the solution of the Burgers equation as defined by (28), with $t=\frac{1}{2}$. The branch points, given for this example by $\pm iy$ with $y=\log(2+\sqrt{3})-\sqrt{3}/2$, are indicated by the white dots on the imaginary axis.

Here we consider a different initial condition, $u_0(x)=-\sin(x)$, which qualitatively leads to the same singularity dynamics (mod 2π) but without the availability of Cardano's formula. Instead, we shall base approximations on the Fourier series solution derived in [Platzman, 1964], namely,

$$u(x,t)=-2\sum_{k=1}^{\infty}c_k(t)\sin(kx),\qquad c_k(t)=\frac{J_k(kt)}{kt}. \tag{26}$$

The J_k denote the Bessel functions. For the purpose of Padé approximation, consider the series

$$f(z,t)=\sum_{k=0}^{\infty}c_{k+1}(t)z^k, \tag{27}$$

which enables one to continue the solution (26) into the complex plane via

$$u(z,t)=ie^{iz}f(e^{iz},t)-ie^{-iz}f(e^{-iz},t). \tag{28}$$

For the particular case $t = \frac{1}{2}$, we truncated the series (27) to 17 terms and approximated it with the $(8, 8)$ linear and $(5, 5, 5)$ quadratic approximants, which were used to compute (28). The resulting approximations were compared with the solution obtained by applying Newton iteration to solve for u from $u = u_0(z - ut)$. The relative differences are shown in Figure 9. As with all the other examples we have seen, the quadratic Padé approximant is significantly better in resolving the underlying function. Note that the region of high accuracy is not localized to a circular region about the origin, as was the case in Figures 3 and 8. Instead, the high accuracy region is spread out in a strip along the real axis, which is typical for Fourier approximations.

We conclude with Table 3, which shows how well the location of the branch point can be determined by these Padé approximants. For the quadratic Padé approximant the branch point location was computed by finding the roots of the discriminant polynomial (7). For the linear Padé approximant we simply selected the pole closest to the true value as given in the caption of Figure 9. The error in the linear approximant decreases algebraically, while in the quadratic case it decreases exponentially.

Table 3. Approximation to the branch point locations shown in Figure 9, with absolute errors in brackets. ℓ is the number of terms used in the series (27)

ℓ	Linear	Quadratic
5	0.7080 (3×10^{-1})	0.461352248460 (1×10^{-2})
11	0.5163 (7×10^{-2})	0.450932810000 (3×10^{-7})
17	0.4803 (3×10^{-2})	0.450932499771 (7×10^{-9})
23	0.4675 (2×10^{-2})	0.450932492988 (2×10^{-10})
exact	0.4509	0.450932493145

7 References

Baker Jr. G. A., Graves-Morris P. Padé approximants. — Second edition. — Cambridge University Press, Cambridge, 1996. — P. xiv + 746.

Bessis D., Fournier J.-D. Complex singularities and the Riemann surface for the Burgers equation // Nonlinear physics (Shanghai, 1989). — Springer, Berlin, 1990. — Res. Rep. Phys. — P. 252–257.

Boyd J. P. Chebyshev expansion on intervals with branch points with application to the root of Kepler's equation: a Chebyshev – Hermite – Padé method // J. Comput. Appl. Math. — 2009. — Vol. 223, No. 2. — P. 693–702.

Corless R. M., Gonnet G. H., Hare D. E. G. et al. On the Lambert W function // Adv. Comput. Math. — 1996. — Vol. 5, No. 4. — P. 329–359.

Corless R., Jeffrey D. Graphing elementary Riemann surfaces // SIGSAM Bulletin. — 1998. — Vol. 32, No. 123. — P. 11–17.

Derksen H. An algorithm to compute generalized Padé – Hermite forms: Tech. rep. / Derksen H.: Report 9403, Katholieke Universiteit Nijmegen, 1994.

Driscoll T., Fornberg B. A Padé-based algorithm for overcoming the Gibbs phenomenon // Numer. Algorithms. — 2001. — Vol. 26, No. 1. — P. 77–92.

Gonnet P., Güttel S., Trefethen L. N. Robust Padé approximation via SVD // SIAM Rev. — 2013. — Vol. 55, No. 1. — P. 101–117.

Gonnet P., Pachón R., Trefethen L. Robust rational interpolation and least-squares // Elect. Trans. Numer. Anal. — 2011. — Vol. 38. — P. 146–167.

Loi S. L., McInnes A. W. An algorithm for the quadratic approximation // J. Comput. Appl. Math. — 1984. — Vol. 11, No. 2. — P. 161–174.

Luke Y. L. Computations of coefficients in the polynomials of Padé approximations by solving systems of linear equations // J. Comput. Appl. Math. — 1980. — Vol. 6, No. 3. — P. 213–218.

Platzman G. W. An exact integral of complete spectral equations for unsteady one-dimensional flow // Tellus. — 1964. — Vol. 16, No. 4. — P. 422–431.

Sergeyev A. V. A recursive algorithm for Padé – Hermite approximations // USSR Comput. Maths. Math. Phys. — 1986. — Vol. 26, No. 2. — P. 17–22.

Shafer R. E. On quadratic approximation // SIAM J. Numer. Anal. — 1974. — Vol. 11. — P. 447–460.

Short L. The evaluation of Feynman integrals in the physical region using multi-valued approximants // J. Phys. G: Nucl. Phys. — 1979. — Vol. 5. — P. 167–198.

Stahl H. The convergence of Padé approximants to functions with branch points // J. Approx. Theory. — 1997. — Vol. 91, No. 2. — P. 139–204.

Stahl H. Spurious poles in Padé approximation // J. Comput. Appl. Math. — 1998. — Vol. 99, No. 1-2. — P. 511–527.